1. Log in to your Sage/Cocalc account.
   
   (a) Start the Chrome browser.
   (b) Go to http://cocalc.com and sign in.
   (c) You should see an existing Project for our class. Click on that.
   (d) Click “New”, call it c11, then click “Sage Worksheet”.

2. **Warm-ups.** What will the following commands return in Sage? Answer and then use Sage to check.
   
   (a) is_prime(245); is_prime(23)
   (b) even = lambda x: x%2==0; even(6); even(7)
   (c) L=[2..10]; len(L)
   (d) [k**2 for k in L]
   (e) n=6
      if n<7:
          print "{} is less than 7".format(n)

   (f) n=6
      if n<5:
          print "{} is less than 7".format(n)

   (g) i=0
      while i<7:
          print i
          i=i+1

3. **Programming.** What will the following code do?
   
   ```python
   def mystery1(n):
       L=[1..n]
       even = lambda x: x%2==0
       print [even(k) for k in L]
   mystery1(9)
   
   def mystery2(n):
       M=[]
       even = lambda x: x%2==0
       for i in [1..n]:
           if even(i)==True:
               M.append(i)
       print M
   mystery2(9)
   ```
A **while loop** runs a block of code while a condition is still satisfied. A common way to use a while loop is in a test where you don’t know precisely when the test condition will be met.

4. What will the following code do?

```python
def mystery3(n):
    M=[]
i=0
even = lambda x: x%2==0
while i<n:
    if even(i)==True:
        M.append(i)
i=i+1
print M
mystery3(9)
```

5. Write a definition for a function that prints the lists [1..i] for i=0 to i=4. Use a while loop. Evaluate and test. Now try to write a definition for a function that prints the lists [1..i] for i=0 to i=n.

6. Now try to write a definition for a function that prints the lists [1..i] for i=0 to i=n. Evaluate and test.

If \( f(x) \) is a continuous function, and \( f(a) \leq c \leq f(b) \) then there is some real number \( x \) in the interval \([a, b]\) where \( f(x) = c \) (that’s the Intermediate Value Theorem). We will define a function that finds this \( x \). We will do this in steps.

7. Given a continuous function \( f(x) \), and numbers \( a \), \( b \) and \( c \), define a function `check_conditions(f,a,b,c)` that returns True if \( f(a) \leq c \leq f(b) \) and False otherwise. Evaluate.

8. Let \( f(x) = x^2 \). Evaluate `check_conditions(f,1,2,3)` and `check_conditions(f,1,2,5)`. Is the output what you expected?

9. Given a continuous function \( f(x) \), and numbers \( a \), \( b \) and \( c \), define a function `test_average(f,a,b,c)` that returns the tuple \((a, (a+b)/2)\) if \( f((a+b)/2) \geq c \) and returns \(((a+b)/2, b)\) if \( f((a+b)/2) < c \).